

Package ‘ConvergenceConcepts’

January 20, 2025

Type Package

Title Seeing Convergence Concepts in Action

Version 1.2.3

Date 2022-08-18

Author Pierre Lafaye De Micheaux [aut, cre],
Benoit Liquet [aut]

Maintainer Pierre Lafaye De Micheaux <lafaye@unsw.edu.au>

Description This is a pedagogical package, designed to help students understanding convergence of random variables. It provides a way to investigate interactively various modes of convergence (in probability, almost surely, in law and in mean) of a sequence of i.i.d. random variables. Visualisation of simulated sample paths is possible through interactive plots. The approach is illustrated by examples and exercises through the function 'investigate', as described in Lafaye de Micheaux and Liquet (2009) <[doi:10.1198/tas.2009.0032](https://doi.org/10.1198/tas.2009.0032)>. The user can study his/her own sequences of random variables.

License GPL (>= 2)

LazyLoad yes

Depends R (>= 2.5.0), tcltk, tkrplot, lattice, grDevices

NeedsCompilation no

Repository CRAN

Date/Publication 2022-08-18 12:10:06 UTC

Contents

check.convergence	2
criterion	3
generate	5
investigate	6
law.plot2d	6
law.plot3d	7
p.as.plot	8
visualize.crit	9
visualize.sp	10

Index**12**

check.convergence	<i>Check convergence</i>
-------------------	--------------------------

Description

This function enables one to investigate the four classical modes of convergence on simulated data: in probability, almost surely, in r -th mean and in law.

Usage

```
check.convergence(nmax,M,genXn,argsXn=NULL,mode="p",epsilon=0.05,r=2,nb.sp=10,
  density=FALSE,densfunc=dnorm,probfunc=pnorm,tinf=-3,tsup=3,plotfunc=plot,...)
```

Arguments

nmax	number of points in each sample path.
M	number of sample paths to be generated.
genXn	a function that generates the X_n-X values, or only the X_n values in the law case.
argsXn	a list of arguments to genXn.
mode	a character string specifying the mode of convergence to be investigated, must be one of "p" (default), "as", "r" or "L".
epsilon	a numeric value giving the interval endpoint.
r	a numeric value ($r>0$) if convergence in r -th mean is to be studied.
nb.sp	number of sample paths to be drawn on the left plot.
density	if density=TRUE, then the plot of the density of X and the histogram of X_n is returned. If density=FALSE, then the plot of the distribution function $F(t)$ of X and the empirical distribution $F_n(t)$ of X_n is returned.
densfunc	function to compute the density of X .
probfunc	function to compute the distribution function of X .
tinf	lower limit for investigating convergence in law.
tsup	upper limit for investigating convergence in law.
plotfunc	R function used to draw the plot: for example plot or points.
...	optional arguments to plotfunc.

Details

The objective of this function is to investigate graphically the convergence of some random variable X_n to some random variable X . In order to use it, you should be able to provide generators of X_n and X (or of X_n-X). The four modes of convergence that you can try are: in probability, almost surely, in r -th mean and in law. For the convergence in law, we compute $\hat{l}_n(t) = |\hat{F}_n(t) - F(t)|$ for ten values equally distributed between t_{inf} and t_{sup} .

Author(s)

P. Lafaye de Micheaux and B. Liqueur

References

Lafaye de Micheaux, P. (<plafaye@club.fr>), Liqueur, B. "Understanding Convergence Concepts: a Visual-Minded and Graphical Simulation-Based Approach", *The American Statistician*, 63:2, 173–178, (2009).

See Also

[criterion](#), [generate](#), [investigate](#), [law.plot2d](#), [law.plot3d](#), [p.as.plot](#), [visualize.crit](#), [visualize.sp](#)

Examples

```
## Not run:

##### Exercise 3 #####
# Let X1, X2, ..., Xn be independent random variables such that #
# P[Xn=sqrt(n)]=1/n and P[Xn=0]=1-1/n #
# Does Xn converges to 0 in 2-th mean? in probability? #
#####

options(example.ask=FALSE)

pnotrgen<-function(n){rbinom(n,1,1/(1:n))*sqrt(1:n)}

check.convergence(nmax=1000,M=500,genXn=pnotrgen,mode="r",r=2)
legend(100,6,legend=expression(hat(e)["n,2"]),lty=1)
tt3.1 <- check.convergence(nmax=1000,M=500,genXn=pnotrgen,mode="p")

## End(Not run)
```

criterion

Convergence criterion computation

Description

This function computes the values of the criterion convergence function for convergence in probability, almost surely or in r -th mean, given the sample paths.

Usage

```
criterion(data,epsilon=0.05,mode="p",r=2)
```

Arguments

data	matrix containing the sample paths of $X_n - X$ values.
epsilon	a numeric value giving the interval endpoint.
mode	a character string specifying the mode of convergence to be investigated, must be one of "p" (default), "as" or "r".
r	a numeric value ($r > 0$) if convergence in r-th mean is to be studied.

Details

The data matrix contains the $X_n - X$ values. If mode="p", criterion approximates $p_n = P[|X_n - X| > \epsilon]$. If mode="as", criterion approximates $a_n = P[\exists k \geq n; |X_k - X| > \epsilon]$. If mode="r", criterion approximates $e_{n,r} = E|X_n - X|^r$. The approximations are based on the frequentist approach.

Value

crit	the vector of criterion values.
------	---------------------------------

Author(s)

P. Lafaye de Micheaux and B. Lique

References

Lafaye de Micheaux, P. (plafaye@club.fr), Lique, B. "Understanding Convergence Concepts: a Visual-Minded and Graphical Simulation Based Approach", *The American Statistician*, 63:2, 173–178, (2009).

See Also

[check.convergence](#), [generate](#), [investigate](#), [law.plot2d](#), [law.plot3d](#), [p.as.plot](#), [visualize.crit](#), [visualize.sp](#)

Examples

```
myrbinom <- function(n,alpha){rbinom(n,1,1/(1:n))*((1:n)**alpha)}
data <- generate(nmax=1000,M=500,myrbinom,args=list(alpha=0.5))$data
critr1 <- criterion(data,mode="r",r=1)$crit
```

generate	<i>Generation of sample paths</i>
----------	-----------------------------------

Description

This function generates the sample paths of a sequence of random variables.

Usage

```
generate(randomgen, nmax=1000, M=500, argsgen=NULL)
```

Arguments

nmax	number of points in each sample path.
M	number of sample paths to be generated.
randomgen	a function that generates the X_n - X values.
argsgen	a list of arguments to randomgen.

Value

data	matrix containing in each row a sample path of X_n - X values.
------	--

Author(s)

P. Lafaye de Micheaux and B. Liqueur

References

Lafaye de Micheaux, P. (<plafaye@club.fr>), Liqueur, B. "Understanding Convergence Concepts: a Visual-Minded and Graphical Simulation-Based Approach", *The American Statistician*, 63:2, 173–178, (2009).

See Also

[check.convergence](#), [criterion](#), [investigate](#), [law.plot2d](#), [law.plot3d](#), [p.as.plot](#), [visualize.crit](#), [visualize.sp](#)

Examples

```
myrbinom<-function(n,alpha){rbinom(n,1,1/(1:n))*((1:n)**alpha)}  
data <- generate(randomgen=myrbinom,argsgen=list(alpha=0.5))$data
```

investigate

Investigate examples and exercises from the paper

Description

This function investigates the convergence for all the examples and exercises from the article cited in references.

Usage

```
investigate()
```

Author(s)

P. Lafaye de Micheaux and B. Liqueur

References

Lafaye de Micheaux, P. (<plafaye@club.fr>), Liqueur, B. "Understanding Convergence Concepts: a Visual-Minded and Graphical Simulation-Based Approach", *The American Statistician*, 63:2, 173–178, (2009).

See Also

[check.convergence](#), [criterion](#), [generate](#), [law.plot2d](#), [law.plot3d](#), [p.as.plot](#), [visualize.crit](#), [visualize.sp](#)

Examples

```
investigate()
```

law.plot2d

Interactive 2D convergence in law

Description

This function enables one to draw the interactive 2D plot to investigate convergence in law.

Usage

```
law.plot2d(data, density=FALSE, densfunc=dnorm, probfunc=pnorm, tinf=-5, tsup=5)
```

Arguments

data	matrix containing the sample paths of X_n values.
density	if density=TRUE, then the plot of the density of X and the histogram of X_n is returned. If density=FALSE, then the plot of the distribution function $F(t)$ of X and the empirical distribution $F_n(t)$ of X_n is returned.
densfunc	function to compute the density of X .
probfunc	function to compute the distribution function of X .
tinf	lower limit for investigating convergence in law.
tsup	upper limit for investigating convergence in law.

Author(s)

P. Lafaye de Micheaux and B. Lique

References

Lafaye de Micheaux, P. (<plafaye@club.fr>), Lique, B. "Understanding Convergence Concepts: a Visual-Minded and Graphical Simulation-Based Approach", The American Statistician, 63:2, 173–178, (2009).

See Also

[check.convergence](#), [criterion](#), [generate](#), [investigate](#), [law.plot3d](#), [p.as.plot](#), [visualize.crit](#), [visualize.sp](#)

Examples

```
rand <- function(n){(cumsum(rchisq(n,df=1))-(1:n))/sqrt(2*(1:n))}
data <- generate(randomgen=rand,nmax=1000,M=500)$data
law.plot2d(data)
```

law.plot3d

Static 3D convergence in law

Description

This function enables one to draw the static 3D plot to investigate convergence in law.

Usage

```
law.plot3d(data,probfunc,tinf=-5,tsup=5)
```

Arguments

data	matrix containing the sample paths of X_n values.
probfunc	function to compute the distribution function of X .
tin	lower limit for investigating convergence in law.
tsup	upper limit for investigating convergence in law.

Author(s)

P. Lafaye de Micheaux and B. Lique

References

Lafaye de Micheaux, P. (<plafaye@club.fr>), Lique, B. "Understanding Convergence Concepts: a Visual-Minded and Graphical Simulation-Based Approach", *The American Statistician*, 63:2, 173–178, (2009).

See Also

[check.convergence](#), [criterion](#), [generate](#), [investigate](#), [law.plot2d](#), [p.as.plot](#), [visualize.crit](#), [visualize.sp](#)

Examples

```
rand <- function(n){(cumsum(rchisq(n,df=1))-(1:n))/sqrt(2*(1:n))}
data <- generate(randomgen=rand,nmax=300,M=2000)$data
law.plot3d(data,pnorm)
```

p.as.plot

Interactive plot for convergence in probability and almost surely

Description

This function enables one to draw the interactive plots used to study convergence in probability and almost surely.

Usage

```
p.as.plot(data,critp,critas,epsilon=0.05,nb.sp=10,mode="p")
```

Arguments

data	matrix containing the sample paths of X_n - X values.
critp	vector containing the criterion values for convergence in probability.
critas	vector containing the criterion values for convergence almost surely.
epsilon	a numeric value giving the interval endpoint.
nb.sp	number of sample paths to be drawn on the left plot.
mode	a character string specifying the mode of convergence to investigate, must be one of "p" (default), "as" or "r".

Author(s)

P. Lafaye de Micheaux and B. Liqueur

References

Lafaye de Micheaux, P. (<plafaye@club.fr>), Liqueur, B. "Understanding Convergence Concepts: a Visual-Minded and Graphical Simulation-Based Approach", *The American Statistician*, 63:2, 173–178, (2009).

See Also

[check.convergence](#), [criterion](#), [generate](#), [investigate](#), [law.plot2d](#), [law.plot3d](#), [visualize.crit](#), [visualize.sp](#)

Examples

```
myrbinom <- function(n,alpha){rbinom(n,1,1/(1:n))*((1:n)**alpha)}
data <- generate(randomgen=myrbinom,nmax=1000,M=500,argsgen=list(alpha=0.5))$data
critp <- criterion(data,epsilon=0.05,"p")$crit
critas <- criterion(data,epsilon=0.05,"as")$crit
p.as.plot(data,critp,critas,epsilon=0.05,nb.sp=10,mode="p")
```

visualize.crit

Visualisation of convergence criterion curves

Description

This function enables one to visualize the convergence criterion curve.

Usage

```
visualize.crit(crit,plotfunc=plot,...)
```

Arguments

crit	vector containing the criterion values.
plotfunc	R function used to draw the plot: for example plot or points.
...	optional arguments to plotfunc.

Author(s)

P. Lafaye de Micheaux and B. Liqueur

References

Lafaye de Micheaux, P. (<plafaye@club.fr>), Liqueur, B. "Understanding Convergence Concepts: a Visual-Minded and Graphical Simulation-Based Approach", *The American Statistician*, 63:2, 173–178, (2009).

See Also

[check.convergence](#), [criterion](#), [generate](#), [investigate](#), [law.plot2d](#), [law.plot3d](#), [p.as.plot](#), [visualize.sp](#)

Examples

```
myrbinom <- function(n,alpha){rbinom(n,1,1/(1:n))*((1:n)**alpha)}
data <- generate(randomgen=myrbinom,nmax=1000,M=500,argsgen=list(alpha=0.5))$data
critr1 <- criterion(data,epsilon=0.05,mode="r",r=2)$crit
visualize.crit(critr1,main="Convergence in 2-th mean?",col="red")
legend(0,5,legend=expression(hat(e)["n,2"]),lty=1,col="red")
```

visualize.sp

Visualisation of sample paths

Description

This function enables one to visualize the sample paths.

Usage

```
visualize.sp(data,epsilon=0.05,nb.sp=10,plotfunc=plot,...)
```

Arguments

data	matrix containing the sample paths of $X_n - X$ values.
epsilon	a numeric value giving the interval endpoint.
nb.sp	number of sample paths to draw on the plot.
plotfunc	R function used to draw the plot: for example plot or points.
...	optional arguments to plotfunc.

Author(s)

P. Lafaye de Micheaux and B. Liqueur

References

Lafaye de Micheaux, P. (plafaye@club.fr), Liqueur, B. "Understanding Convergence Concepts: a Visual-Minded and Graphical Simulation-Based Approach", *The American Statistician*, 63:2, 173–178, (2009).

See Also

[check.convergence](#), [criterion](#), [generate](#), [investigate](#), [law.plot2d](#), [law.plot3d](#), [p.as.plot](#), [visualize.crit](#)

Examples

```
myrand <- function(n,...){cumsum(rnorm(n,...))/(1:n)}  
data <- generate(randomgen=myrand,nmax=1000,M=500)$data  
visualize.sp(data,main="Ten sample paths",col="grey")
```

Index

* **distribution**

- check.convergence, 2
- criterion, 3
- generate, 5
- investigate, 6
- law.plot2d, 6
- law.plot3d, 7
- p.as.plot, 8
- visualize.crit, 9
- visualize.sp, 10

* **dplot**

- check.convergence, 2
- investigate, 6
- law.plot2d, 6
- law.plot3d, 7
- p.as.plot, 8
- visualize.crit, 9
- visualize.sp, 10

* **htest**

- check.convergence, 2
- criterion, 3
- investigate, 6
- law.plot2d, 6
- law.plot3d, 7
- p.as.plot, 8
- visualize.crit, 9
- visualize.sp, 10

* **iplot**

- check.convergence, 2
- investigate, 6
- law.plot2d, 6
- p.as.plot, 8

check.convergence, 2, 4–10

criterion, 3, 3, 5–10

generate, 3, 4, 5, 6–10

investigate, 3–5, 6, 7–10

law.plot2d, 3–6, 6, 8–10

law.plot3d, 3–7, 7, 9, 10

p.as.plot, 3–8, 8, 10

visualize.crit, 3–9, 9, 10

visualize.sp, 3–10, 10